Creating Real Data for Machine Learning

Build a smartphone app for collecting movement data

Why use an app?

We wanted to show a use case that is often requested from our clients, use it to create real data, and show our machine learning model “learning” in real time.

There are many different sensors you can purchase for detecting movement or vibration, but most people already have one right on them! Almost all smartphones have an accelerometer and many also have a gyroscope.

All apps (and even the browser in many cases) have access to the precise orientation of a phone to adjust displays, measure steps taken, etc.

We will be going through the creation of an application that takes all of this movement data and pushes it to the cloud. With that data, we’ll use machine learning tools to show a real world example of anomaly detection, movement classification, and real time graphing.

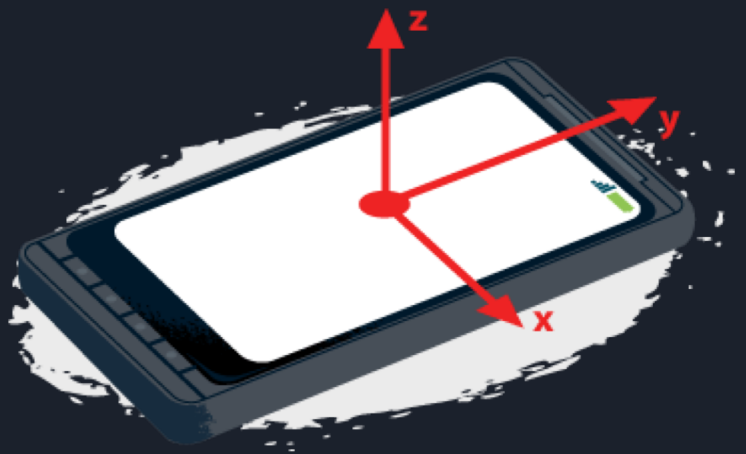
The Sensors

Coordinate system

X - In the plane of the screen, positive to the right.

Y - In the plane of the screen, positive towards the top.

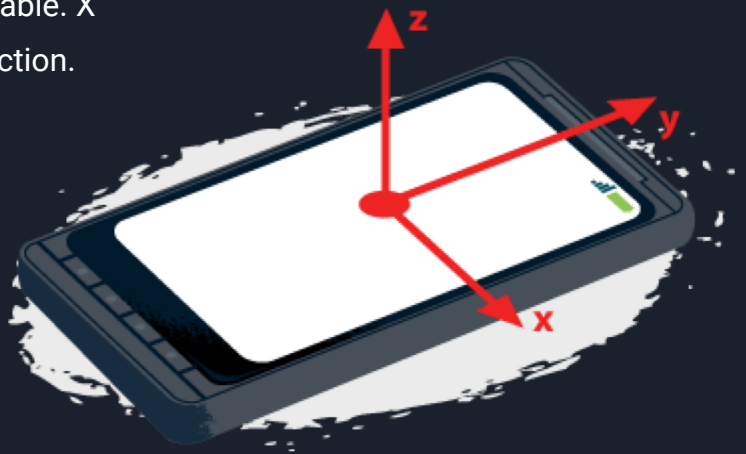
Z - Perpendicular to the screen or keyboard, positive extending away.



https://developers.google.com/web/fundamentals/native-hardware/device-orientation/

Accelerometer

Acceleration is a measurement of change in movement of an object. If the data is not adjusted for the effect of gravity, it will show a value of about 9.8 for the Z axis if sitting on a table. X and Y will be 0 until you tip the phone in a different direction.



Note: There is no current standard for device acceleration values. Each axis can be positive or negative, and some devices will correct for the effect of gravity.

https://developers.google.com/web/fundamentals/native-hardware/device-orientation/

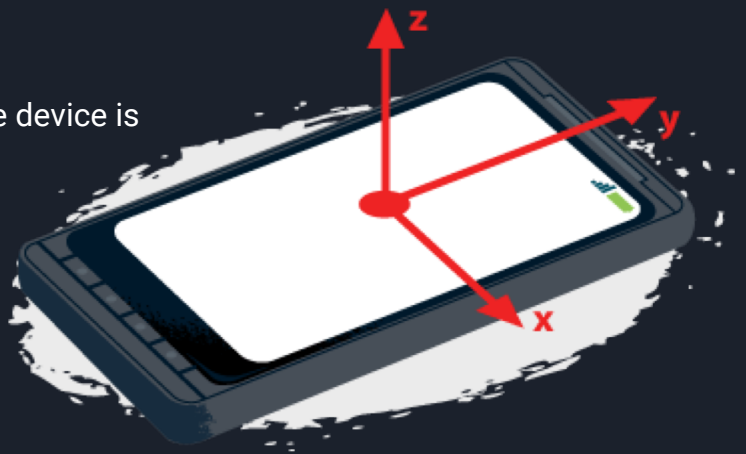
Gyroscope (not all devices)

The gyroscope measure rotation of the device. The phone reports data in degrees, and 0° represents a fixed point in space that is the same for each device:

Alpha: rotation around the z axis, 0° when the top of the device is pointed directly north

Beta: rotation around the x axis, 0° when the top and bottom of the device are equidistant from the surface of the earth

Gamma: rotation around the y axis, 0° when the left and right edges of the device are equidistant from the surface of the earth



https://developers.google.com/web/fundamentals/native-hardware/device-orientation/

Accessing the Data

Two approaches:

1. Build a native app! The native layer of devices have access to accelerometer and gyroscope data in the background, which could be very useful as a tool when the user is not necessarily engaged with the device.
2. Build a web app! These data are available in most modern browsers on a device, and they require no special permissions. Though they're not available when the screen is off, a web app makes it easy to get everyone sending lots of data.

Today we'll walk through the steps of setting up a React Native App, and using the sensor information available at the native level to push data to the cloud. Then we'll experiment with a web app as a group and see if our machine learning setup in the cloud can detect some simple gestures!

Getting Started with React Native

https://facebook.github.io/react-native/docs/getting-started.html

Follow instructions in Building Projects with Native Code tab.

For today, if you're looking to jump right in, you'll need to at least have Node.js and Android Studio or Xcode code already properly installed. Depending on what you do, you may need Homebrew for MacOS, and you may want to be familiar with git. If this means nothing to you, don't worry! We want everyone to be able to follow along and get ideas no matter what.

If you have all the requirements installed, install React Native CLI

```
npm install -g react-native-cli
```

You're ready to start a project!

Create the app and add the sensor access via npm

Initialize a new project

```
react-native init sensorApp
```

Install React Native Sensors and link it to the project automatically:

```
npm install react-native-sensors --save
```

```
react-native link react-native-sensors
```

Attach the sensor info to the display and send to the cloud

Link up the user interface and now you have a sensing device that displays data and sends it to the cloud.

Try running it on your device

```
cd sensorApp
react-native run-android
```

Other cool sensors that may available on your smartphone:

Ambient Light Sensor - used for screen brightness settings

Barometer - to detect atmospheric pressure, makes GPS more accurate by estimating distance from sea level

Proximity - used to detect nearby objects, like your ear when you're talking on the phone

Thermometer - internal phone temp and external temp

Magnetometer - a compass, adds to the accuracy of the accelerometer and gyroscope

Then, there's the ones you're probably really familiar with that could be used to do some great machine learning projects: GPS, Camera and Microphone!

Conclusion

Me: Annie Tomassoni : annie@lab651.com feel free to contact me with questions!

The git repo for the React Native app in this presentation

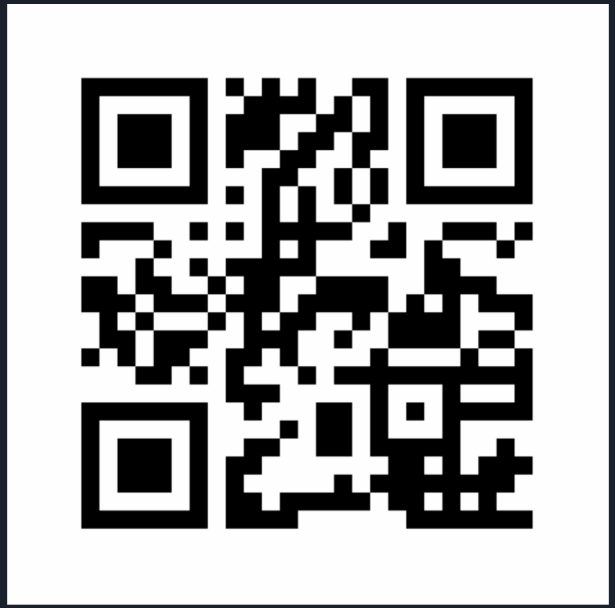
https://github.com/lab651/accelerometer-rn

The repo and link to the web app we'll use for data collection

Repo: https://github.com/lab651/accelerometer-pwa

Link: http://bit.ly/2r1A7Ev or https://d3bj9y57758kgm.cloudfront.net/

App for data collection



http://bit.ly/2r1A7Ev